

Wybrany temat — system uczelniany — jest naturalnie grafowy, ponieważ jego elementy (studenci, prowadzący, kierunki, przedmioty, egzaminy, oceny) są silnie powiązane relacjami. Drugim, powiązanym zbiorem danych został system biblioteczny, który umożliwia odwzorowanie relacji pomiędzy studentami, książkami, autorami i przedmiotami (np. książki polecane do kursów).

Połączenie obu systemów (USOS Biblioteka) pozwoliło stworzyć wielowarstwowy graf liczący ponad 200 węzłów i kilkaset relacji, obejmujący 10 typów węzłów i 12 typów krawędzi.

W projekcie wykorzystano następujące typy węzłów: Student, Prowadzący, Przedmiot, Kierunek, Wydział, Egzamin, Grupa, Ocena, Książka, Autor.

Przykładowe relacje: STUDIUJE_NA, PROWADZI, UCZESTNICZY_W, ZDOBYŁ_OCENĘ, EGZAMIN_Z, WYPOŻYCZYŁ, POLECANA_DO, NAPISAŁ, WYDAŁO.

Zaimplementowana procedura:

```
MATCH (s:Student)-[:ZDOBYŁ_OCENĘ]->(o:Ocena)
WITH s, avg(o.wartość) AS średnia
SET s.średnia_ocen = round(średnia,2);
```

Procedura oblicza i zapisuje średnie oceny studentów, pokazując praktyczne zastosowanie przetwarzania danych w grafie.

Przygotowano 8 zapytań analitycznych, w tym:

- zapytania z agregacją i sortowaniem (np. ranking studentów i przedmiotów),
- zapytania z użyciem UNION (porównanie liczby studentów na kierunkach i wydziałach),
- zapytania z MERGE (tworzenie relacji KOLEGA_Z_GRUPY między studentami),
- zapytania łączące dane z obu systemów (np. książki wypożyczone przez studentów określonych kierunków).

W bazie utworzono indeksy dla często wyszukiwanych pól, takich jak imię studenta czy nazwa przedmiotu. Dzięki temu wyszukiwanie danych stało się znacznie szybsze. Po uruchomieniu polecenia PROFILE w Neo4j można było zobaczyć, że zapytania korzystają z indeksu (NodeIndexSeek), a nie skanują całej etykiety (NodeByLabelScan). W praktyce skróciło to czas wykonywania zapytań kilkukrotnie.

W analizie grafu można zauważyć, że węzeł Student pełni rolę łącznika (mostu) między częścią uczelnianą i biblioteczną. Podobnie węzeł Przedmiot łączy prowadzących, grupy i książki polecane do nauki. Można wyróżnić kilka podgrafów: akademicki (studenci, przedmioty, egzaminy), biblioteczny (książki, autorzy, biblioteki) oraz administracyjny (wydziały, kierunki).

Projekt pokazał, że bazy grafowe są bardzo przydatne w modelowaniu złożonych zależności. Zastosowanie Neo4j umożliwiło przedstawienie powiązań w sposób czytelny i elastyczny, a przygotowane zapytania pozwalają łatwo analizować dane i odkrywać zależności pomiędzy elementami systemu.

Cały system spełnia wymagania projektowe: posiada dużą, połączoną strukturę grafu, indeksy, procedury, nietrywialne zapytania oraz skrypt odtwarzający całą bazę danych. Projekt można w prosty sposób rozszerzyć o kolejne moduły, np. system dziekanatowy lub platformę e-learningową.